LLM-Modulo Frameworks as Compound AI Architectures for Robust Planning

Subbarao Kambhampati * **Karthik Valmeekam** **Lin Guan** **Mudit Verma**
Siddhant Bhambri **Kaya Stechly** **Lucas Saldyt** **Atharva Gundawar**
School of Computing & AI, Arizona State University

(Extended abstract of `https://arxiv.org/abs/2402.01817`, to appear at ICML 2024)

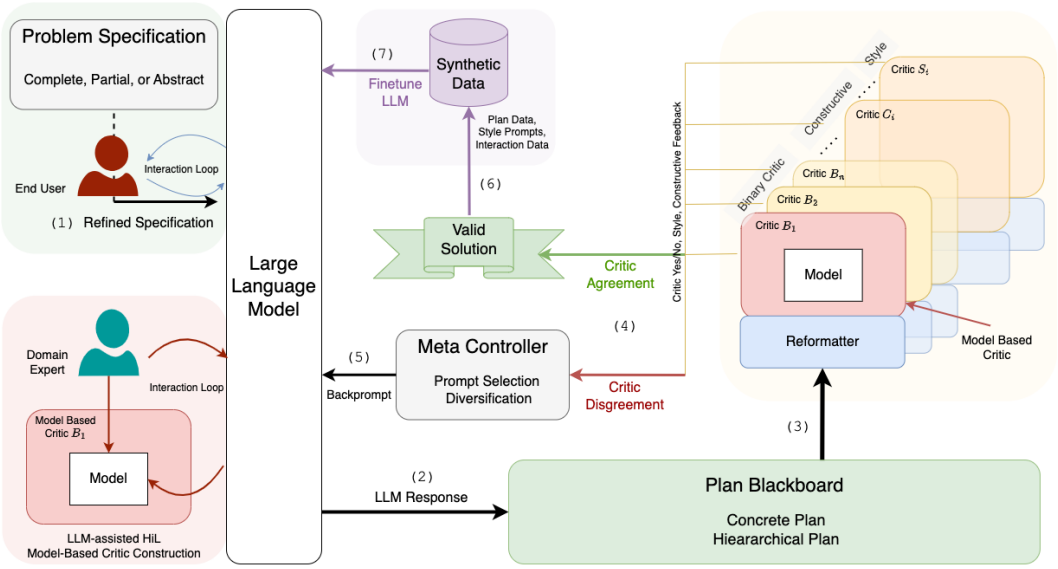

Figure 1: The proposed LLM-Modulo framework where LLMs act as idea generators and various external critics that specialize in different aspects, critique the candidate plan.

The versatility of LLMs has led many researchers to wonder whether they can also do well on planning and reasoning tasks typically associated with System 2 competency. On the face of it, this doesn't seem to ring true, as both by training and operation, LLMs are best seen as a giant pseudo System 1 (Kahneman, 2011). Not surprisingly, initial excitement based on anecdotal performance of LLMs on reasoning tasks (Bubeck et al., 2023) has dissipated to some extent by the recent spate of studies questioning the robustness of such behaviors–be it planning (Valmeekam et al., 2023b; Kambhampati, 2024), simple arithmetic and logic (Dziri et al., 2023), theory of mind abilities (Ullman, 2023; Verma et al., 2024), or general mathematical and abstract benchmarks (McCoy et al., 2023; Gendron et al., 2023). Despite this, a steady stream of claims continue to be made in the literature about the planning and reasoning capabilities of LLMs. Of particular concern is the head long rush to create and deploy the so-called *agentic LLM systems*, which conflate the flexibility with which LLMs can invoke external services ("actions") with their planning abilities. Acting without planning abilities should raise paramount AI Safety concerns!

The fact that LLMs lack robust planning capabilities in autonomous modes *doesn't however imply that LLMs don't have any constructive roles to play in solving planning/reasoning tasks*. On the contrary, the fact that they are approximate knowledge sources for everything including planning,

*Corresponding author. Email: rao@asu.edu

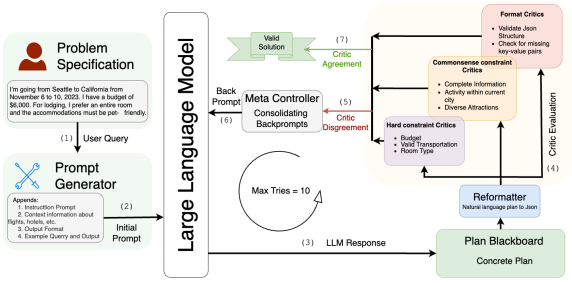

Figure 2: LLM Modulo Framework adapted for Travel Planning

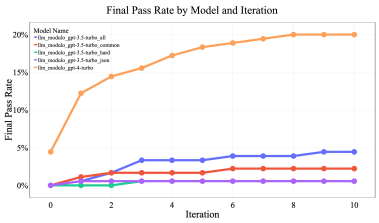

Figure 3: Final Pass rates of models across LLM Modulo Iterations

can be leveraged in a generate-test-critique setups in conjunction with either model-based verifiers or expert humans in the loop. Accordingly, we propose a general "**LLM-Modulo**" framework (see Figure 1), our own version of a *compound AI system* that can leverage LLMs to provide flexible and robust planning capabilities. As can be seen from the figure, the underlying architecture is a Generate-Test-Critique loop. The loop starts with the LLM getting the problem specification and generating its first plan candidate (step 2 in the figure). The *bank of critics* then critique the candidate both for correctness and style. These critiques are collated (see *meta controller*), and sent as a "back prompt" to the LLM to generate the next candidate. In addition to its primary role of guessing/generating candidate plans in response to the specifcation and back prompts from the critics, LLMs play a spectrum of other important roles in this architecture. To begin with, LLMs can help in pooling and diversifying the back prompts from the critics (as part of the *meta controller*). Secondly, the LLM plays a role in converting the guessed plan candidate into specialized representations used by the various critics (e.g., the time-line view, the causal link view etc.). This *reformatter* role leverages the fact that LLMs are very good at format conversion (c.f. (Olmo et al., 2021)) Third, LLM also plays a role in helping the end user flesh out the incomplete problem specification to begin with (Step 1 in Figure 1). Finally, while LLMs can't serve as correctness critics (c.f. (Valmeekam et al., 2023a; Stechly et al., 2023)), they can be effective as style critics (Guan et al., 2024). Even for the correctness critics, the LLM can play a role in helping the domain expert tease out and refine the domain models used by the various model-based critics (Guan et al., 2023; Kwon et al., 2022) (red box on bottom left).

All this leveraging of LLMs is done without ascribing to them any guaranteed planning or correctness verification abilities. The LLM ideas are vetted by external critics, thus ensuring that the plans generated in this architecture can have formal correctness guarantees where possible. Compared to traditional model-based planners that are guaranteed to be correct in a narrow set of domains, LLMs may likely be good at generating plausible (but not guaranteed to be correct) plan heuristics/suggestions in many more scenarios, regardless of the computational complexity of the underlying planning problem class. Thus, LLM-Modulo architecture allows for more flexibility than the traditional planning architectures studied in AI (Ghallab et al., 2004), which put *a priori* constraints on the expressiveness of the problems that can be posed to the planner.

Application of LLM-Modulo Architecture for Travel Planning: A real-world use case of the LLM-Modulo architecture is illustrated by our experiments with the TravelPlanning challenge by (Xie et al., 2024). Authors of the benchmark themselves provide variations of agentic LLMs such as GPT models paired with prompt engineering techniques like Chain of Thought and ReAct, and report that the best LLM strategies exhibit a startlingly low 0.6% performance rate! As described fully in (Gundawar et al., 2024), we adapt our LLM Modulo architecture and operationalize their hard-constraints (such as budget constraint set by the user) or common-sense constraints (such as suggesting diverse attractions to visit) as critics as in Figure 2. Our preliminary results show (see Figure 3) that LLM-Modulo based agentification with automated critics in the loop significantly improves the performance (6x of baselines) even with alimit of 10 back prompting cycles, and weaker models such as GPT-3.5-turbo. Furthermore, we also find that LLMs can successfully implement functions corresponding to hard critics and several common-sense critics. Finally, LLMs reliably play the role of reformatter as well, converting free form travel plans into structured plans parseable by the critics for backprompts or plan evaluation.

References

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Dziri, N., Lu, X., Sclar, M., Li, X. L., Jiang, L., Lin, B. Y., Welleck, S., West, P., Bhagavatula, C., Bras, R. L., Hwang, J. D., Sanyal, S., Ren, X., Ettinger, A., Harchaoui, Z., and Choi, Y. Faith and fate: Limits of transformers on compositionality. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=Fkckkr3ya8.

Gendron, G., Bao, Q., Witbrock, M., and Dobbie, G. Large language models are not abstract reasoners. *arXiv preprint arXiv:2305.19555*, 2023.

Ghallab, M., Nau, D., and Traverso, P. *Automated Planning: theory and practice*. Elsevier, 2004.

Guan, L., Valmeekam, K., Sreedharan, S., and Kambhampati, S. Leveraging pre-trained large language models to construct and utilize world models for model-based task planning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=zDbsSscmuj.

Guan, L., Zhou, Y., Liu, D., Zha, Y., Amor, H. B., and Kambhampati, S. "task success" is not enough: Investigating the use of video-language models as behavior critics for catching undesirable agent behaviors, 2024.

Gundawar, A., Verma, M., Guan, L., Valmeekam, K., Bhambri, S., and Kambhampati, S. Robust planning with llm-modulo framework: Case study in travel planning. *arXiv preprint arxiv:2405.20625*, 2024.

Kahneman, D. *Thinking, fast and slow*. macmillan, 2011.

Kambhampati, S. Can LLMs reason and plan? *Annals of the New York Academy of Sciences*, 2024.

Kwon, M., Xie, S. M., Bullard, K., and Sadigh, D. Reward design with language models. In *The Eleventh International Conference on Learning Representations*, 2022.

McCoy, R. T., Yao, S., Friedman, D., Hardy, M., and Griffiths, T. L. Embers of autoregression: Understanding large language models through the problem they are trained to solve. *arXiv preprint arXiv:2309.13638*, 2023.

Olmo, A., Sreedharan, S., and Kambhampati, S. Gpt3-to-plan: Extracting plans from text using gpt-3. *FinPlan 2021*, pp. 24, 2021.

Stechly, K., Marquez, M., and Kambhampati, S. GPT-4 Doesn't Know It's Wrong: An Analysis of Iterative Prompting for Reasoning Problems. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2023.

Ullman, T. Large language models fail on trivial alterations to theory-of-mind tasks. *arXiv preprint arXiv:2302.08399*, 2023.

Valmeekam, K., Marquez, M., and Kambhampati, S. Can large language models really improve by self-critiquing their own plans? In *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2023a.

Valmeekam, K., Marquez, M., Sreedharan, S., and Kambhampati, S. On the planning abilities of large language models - a critical investigation. In *Thirty-seventh Conference on Neural Information Processing Systems (Spotlight)*, 2023b. URL https://openreview.net/forum?id=X6dEqXIsEW.

Verma, M., Bhambri, S., and Kambhampati, S. Theory of mind abilities of large language models in human-robot interaction: An illusion? *arXiv preprint arXiv:2401.05302*, 2024.

Xie, J., Zhang, K., Chen, J., Zhu, T., Lou, R., Tian, Y., Xiao, Y., and Su, Y. Travelplanner: A benchmark for real-world planning with language agents. *arXiv preprint arxiv:2402.01622*, 2024.